This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNNLS.2025.3542719, IEEE Transactions on Neural Networks and Learning Systems

Fusionformer: A Novel Adversarial Transformer Utilizing Fusion Attention for Multivariate Anomaly Detection

Chuang Wang, Zidong Wang, *Fellow*, *IEEE*, Hongli Dong, *Senior Member*, *IEEE*, Stanislao Lauria, Weibo Liu, Yiming Wang, Futra Fadzil, and Xiaohui Liu

***Abstract*—Multivariate time series forecasting (MTSF) is of significant importance in the enhancement and optimization of real-world applications. The task of MTSF poses substantial challenges due to the unpredictability of temporal patterns and the complexity in modeling the influence of all non-predictive sequences on the target sequence at different time stages. Recent research has demonstrated the potential held by the Transformer algorithm to augment long-term forecasting capability. However, certain obstacles considerably obstruct the direct application of the Transformer to MTSF, such as an unsuitable embedding method, inadequate consideration of inter-variable associations, and the intrinsic restriction of the point-wise objective function. To overcome these challenges, the Fusionformer, an effective Transformer-based forecasting model, is put forth in this paper, which is characterized by three distinctive features: (1) the introduction of a segment-wise sequence embedding method allows for the conversion of the input sequence into multiple informative segments; (2) the implementation of a fusion attention mechanism, designed to capture predominant features across the time dimension and to model intricate inter-variable dependencies; and (3) the development of an adversarial learning method, equipped with an auxiliary discriminator, facilitates the learning of data distribution, instead of progressively correcting the prediction error, thus substantially enhancing the MTSF's accuracy. Furthermore, a Fusionformer-based risk assessment (FRA) method is structured for open-pit mine slope failure early warning issue (SFEW), which aims to prevent potential disasters by accurately predicting future slope movement trends and assessing the probabilities of landslide occurrences. Experimental outcomes validate that Fusionformer outperforms existing forecasting methods, while the FRA framework provides valuable insights and practical guidance for real-world applications.**

***Index Terms*—Fusionformer, adversarial learning, multivariate time series forecasting, slope failure, digital mine**

I. Introduction

Multivariate time series (MTS) data, characterized by multiple dimensions with each dimension representing a specific univariate time series, is widely observed in diverse fields, such as medicine, finance, and industrial applications [4], [13], [22], [42]. MTS forecasting (MTSF), a discipline seeking to predict the future values of MTS based on historical records, has been accorded significant importance in various sectors, including electricity consumption, traffic prediction, and the prediction of remaining useful life [9], [16]. In addition to the aforementioned applications where predicted values are directly utilized, MTSF also aids in decision-making processes for various downstream tasks. Examples include monitoring the stability of open-pit mine slopes and industrial fault surveillance [8], [47]. Consequently, MTSF holds a critical position in the automation and optimization of intelligent real-world scenarios.

Accomplishing an accurate MTSF poses a substantial challenge, as it requires the joint modeling of both intra-variable dependencies (i.e., correlations between different time points within a single variable) and inter-variable dependencies (i.e., correlations between sequences belonging to different variables). In recent years, the swift progression of deep learning has led to the emergence of numerous neural network models, which have empirically demonstrated superior performance in MTSF [14], [21], [26], [27], [32]. One group of work treats the MTS input as a vector sequence and employs convolutional neural network (CNN)/recurrent neural network (RNN) and their variants to capture temporal dependencies [3], [17], [54]. For instance, a temporal convolutional network (TCN) has been proposed in [3] to capture long-range temporal patterns by integrating causal convolutions, dilated convolutions, and skip-connections. In [17], a multiple nested long short-term memory (LSTM) network has been developed to infer inter-relationships between different dimensions and highly volatile pattern changes. A different strand of work, known as graph neural network (GNN), perceives each variable in the MTS as a node in the graph, with dependencies between variables represented as edges. As an example, the Transformer algorithm with edge-enhanced dynamic graphs has been used to capture

This work was supported in part by the European Union's Horizon 2020 Research and Innovation Programme under Grant 869529 (DIG_IT), the National Natural Science Foundation of China under Grants 62403119 and U21A2019, the Hainan Province Science and Technology Special Fund of China under Grant ZDYF2022SHFZ105, the Postdoctoral Fellowship Program of CPSF under Grant Number GZB20240136, the China Postdoctoral Foundation under Grant Number 2024MD753911, the Engineering and Physical Sciences Research Council (EPSRC) of the UK, the Royal Society of the UK, and the Alexander von Humboldt Foundation of Germany. *(Corresponding author: Hongli Dong.)*

Chuang Wang and Hongli Dong are with the Artificial Intelligence Energy Research Institute, Northeast Petroleum University, Daqing 163318, China, also with the Heilongjiang Provincial Key Laboratory of Networking and Intelligent Control, Northeast Petroleum University, Daqing 163318, China, and also with Sanya Offshore Oil & Gas Research Institute, Northeast Petroleum University, Sanya 572025, China. (Emails: wangchuang64@126.com, shiningdhl@vip.126.com)

Zidong Wang, Stanislao Lauria, Weibo Liu, Futra Fadzil and Xiaohui Liu are with the Department of Computer Science, Brunel University London, Uxbridge, Middlesex, UB8 3PH, United Kingdom. (Email: Zidong.Wang@brunel.ac.uk)

Yiming Wang is with the School of Psychology and Neuroscience, University of Glasgow, Glasgow, G12 8QQ, United Kingdom. (Email: yiming.wang@glasgow.ac.uk)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNNLS.2025.3542719, IEEE Transactions on Neural Networks and Learning Systems

temporal associations and dimensional dependencies in [28]. In [4], a multi-scale adaptive GNN has been proposed to learn the multi-scale temporal patterns of the MTS.

Despite deep learning having established itself as the state-of-the-art (SOTA) method for MTSF due to its powerful representational capability, two critically important issues have not been adequately addressed yet. Firstly, existing studies [38], [40] do not consider intra- and inter-variable associations separately (e.g. using convolutional networks to capture dependencies directly from multivariate time-series matrices), which may bias the models toward learning association patterns among different variables from an ambiguous perspective. In reality, different variables within the MTS embody various influential factors of the problem under study, with each possessing a unique data distribution and temporal pattern. In real-world scenarios, each sequence type is generally influenced by multiple relevant variables, each to varying degrees. For instance, in open-pit mining operations, the incidence of landslides depends not solely on actual orebody movement, but also on the combined effect of diverse variables such as temperature and rainfall. As such, it becomes crucial to focus on modeling the association weights among different variables for accurate prediction, a perspective largely overlooked in prior works.

Another issue with existing methods is their typical optimization of a specific statistical metric (such as mean squared error (MSE), mean absolute error (MAE), or likelihood loss) to minimize the discrepancy between the ground truth values and the predicted values on a point-by-point basis. However, such point-wise objectives find it challenging to model the stochastic behavior inherent in real-world time series, leading to unreliable trends in predicted future data [45]. For instance, the stability of a slope in an open-pit mine can be influenced simultaneously by changes in weather and geological conditions, without a clear periodic pattern. Moreover, MSE strives to achieve a global optimum by optimizing the mean of all possible outcomes of the model, which could potentially result in the loss of fine-grained features of the time series. Therefore, an effective MTSF model should be capable of learning highly variable temporal patterns within the MTS.

Motivated by the discussions above, this paper presents the Fusionformer, an adversarial learning-based Transformer equipped with a Fusion Attention Mechanism (FAM). Firstly, a Segment-Wise Sequence Embedding (SWSE) method is introduced, wherein the input sequences are partitioned into multiple segments and embedded into a two-dimensional vector array, corresponding to time and factor. Subsequently, the FAM, which comprises a Segment-Wise Intra-Variable Attention (SWAA) and Segment-Wise Inter-Variable Attention (SWEA), is proposed to amalgamate information about temporal dependencies within a single variable and the interaction effects among different variables. Following this, an auxiliary discriminator is introduced to learn comprehensive representations of the historical sequences and to shape the distributions of the predicted values. The adversarial learning process between the discriminator and the Fusionformer nudges the predictions towards regions of the solution space that have a high probability of containing realistic features, which helps to overcome the weaknesses of the point-wise single objective function and to enhance the inference accuracy at the sequence level. Equipped with adversarial learning and FAM, the Fusionformer is capable of learning valid temporal patterns and association weights from historical time series, thereby accomplishing an accurate MTSF.

Slope failure accidents are among the most common hazards in open-pit mining operations, leading to significant threats to human lives and property [7], [30], [58]. Consequently, it is vital to provide reliable early warnings of progressive slope movements for potentially unstable mines, offering crucial response time for each hazard event. Early warning issue of slope failure (SFEW) is a challenging but pivotal aspect of intelligent mining, with the aim of preventing disasters by predicting future slope movement trends and assessing landslide occurrence probabilities. With the rapid advancements in artificial intelligence, deep neural network-based MTSF models have emerged as promising tools for a range of prediction tasks. This has inspired us to propose the Fusionformer-based risk assessment (FRA) method to enhance the accuracy of SFEW.

The overarching aim of SFEW is to guarantee the safety and efficiency of mining operations, which involves key steps such as slope state prediction and failure hazard assessment. The former aims to predict potential changes in the slope state, while the latter seeks to calculate failure probabilities based on these predictions. Specifically, the proposed Fusionformer is first utilized to learn prominent event patterns over time and intricate correlations among different variables from the input MTS data, with the aim of predicting the future trend of slope movement. Following this, a FRA method is proposed to translate the predicted slope state into a failure probability, where the evolutionary patterns of target historical sequences can be effectively transferred to future time points, leveraging the valuable insights drawn from other sequences.

The principal contributions of this paper can be summarized as follows:

1) We propose the FAM to learn specific evolutionary trends of different dimensional sequences effectively, by fusing intra-variable and inter-variable dependencies.
2) An adversarial learning method is introduced into the Fusionformer training to tackle the stochastic nature of real-world time series, thereby further enhancing the precision of the MTSF at the sequence level.
3) We develop the FRA method for failure hazard assessment, introducing a learnable scaling parameter of the Gaussian kernel to assess the failure degree, and devising a new criterion to calculate the failure probability.
4) Extensive experiments are conducted on four real-world open-pit mine datasets. The experimental results reveal that our proposed Fusionformer algorithm outperforms some state-of-the-art MTSF algorithms, thereby further demonstrating the efficacy and efficiency of our algorithm in addressing SFEW tasks.

The remainder of this paper is organized as follows. Sections II and III offer an overview of related works and preliminaries, respectively. Section IV provides a detailed description of the novel Fusionformer algorithm. Experimental results and

pertinent analysis are presented in Section V, while Section VI concludes the paper.

II. Related Work

A. MTSF

Early approaches to MTSF are grounded in statistical methods. For instance, a forecasting method that integrates the eigenvalue decomposition of the Hankel matrix with an autoregressive integrated moving average (ARIMA) model has been proposed in [31] to handle non-stationary time series. In [1], the ARIMA model has been combined with the Particle Swarm Optimization (PSO) method to enhance forecasting performance.

Recently, many deep neural networks have empirically outperformed statistical ones due to their capabilities in modeling non-linear dependencies. For example, a hierarchical correlation pooling boosted GNN has been proposed in [42] to learn hierarchical relationships and dynamic properties from MTS data. In [55], a hybrid deep neural network was designed to capture coupled and non-linear dynamic features within the MTS, introducing a Deep Convolutional Neural Network (DCNN) and a Gated Recurrent Unit (GRU). Furthermore, an LSTM-based temporal change information learning method has been developed in [57] to capture change information over time from the error gradient flow. Moreover, in [49], a deep hybrid network has been proposed to reduce data uncertainty for accurate MTSF by employing CNN and bidirectional GRU. Nevertheless, these methods have a limited capacity to learn indeterminate temporal patterns and hierarchical variable dependencies, which hampers the advancement of MTSF.

The Transformer, introduced in [34], has achieved significant performance in sequential data processing applications such as machine translation [48], speech recognition [5], and action recognition [33]. The Transformer's capability for modeling long-term dependencies has recently caught the attention of researchers in the MTSF domain. As a result, a series of studies have been reported, see e.g. [56], [59], [60]. In [56], a model called Crossformer has been proposed that captures dependencies across time and dimensions using two-stage attention layers. In [59], a model named Informer has been introduced to enhance prediction performance by employing a ProbSparse self-attention mechanism and incorporating a distilling operation. Lastly, in [60], a frequency enhanced decomposed Transformer has been developed to learn the overall trend and fine-grained features of the time series by integrating the seasonal-trend decomposition method with the Transformer.

III. Preliminary

A. Problem Formulation

Generally, let the historical time series $\mathcal{X}_{1:T} = \left\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T \mid \mathbf{x}_t \in \mathbb{R}^D\right\}$ be given, where T is the length of look-back window and $D > 1$ is the number of variables. MTSF aims to forecast corresponding sequence $\mathcal{Y}_{T+1:T+\tau} = \left\{\mathbf{y}_{T+1}, \ldots, \mathbf{y}_{T+\tau} \mid \mathbf{y}_t \in \mathbb{R}^D\right\}$ within a defined future horizon τ. Therefore, the MTSF model can be formulated as follows:

$$\hat{\mathcal{Y}}_{T+1:T+\tau} = f\left(\mathcal{X}_{1:T}; \Theta\right), \tag{1}$$

where f is the prediction model with learnable parameters Θ, and $\hat{\mathcal{Y}}_{T+1:T+\tau}$ is a set of predicted time points.

IV. Methodology

A. Motivations

In this section, we will analyze the weakness of existing MTSF models as well as anomaly detection models and provide a preview of the Fusionformer.

The Transformer, an encoder-decoder-based model, leverages a multi-head self-attention layer to effectively model long-term dependencies from various perspectives. Within the Transformer, intra-variable dependencies can be modeled using a self-attention map, which displays the distribution of association weights for all data points along the time dimension. This distribution offers a comprehensive depiction of the context and dynamic properties of the time series. However, when applied to MTSF, the Transformer model presents three significant limitations.

1) The Transformer, initially developed for natural language processing (NLP), typically maps a sequence of words into a digital input using learned embeddings, where each word is assigned to a vector of dimension d_{model}. Many existing Transformer-based MTSF models adopt the same embedding approach as NLP. However, unlike information-rich words, a single data point at a specific timestep in the MTS lacks substantial information. This makes it less meaningful and ineffective for improving prediction performance.
2) Recent Transformer-based MTSF methods aim to model variable dependencies by embedding data points from all dimensions into a feature vector and calculating the association weights among different time steps. However, these methods primarily focus on learning intra-variable dependencies rather than inter-variable dependencies, which may limit their forecasting capabilities.
3) Most Transformer-based MTSF methods prioritize optimizing a point-wise loss function that only corrects prediction errors within a fixed length. However, such loss functions fail to learn long-term non-deterministic dynamic variations from the entire series, which may subsequently result in degraded prediction performance.

To address these issues, we propose a simple yet effective model, referred to as Fusionformer, which builds upon the encoder-decoder Transformer and incorporates adversarial learning. The Fusionformer comprises three key components: 1) the SWSE method which aims to provide informative segments for model training; 2) the FAM method which seeks to integrate intra- and inter-variable dependencies; and 3) the adversarial learning method aiming to learn the dynamic patterns of the MTS and accurately predict future values.

The challenge of time series anomaly detection lies in the need to learn effective representations from complex temporal dynamics, while simultaneously developing a criterion to distinguish rare anomalies from the abundance of normal time points. Various classic methods, such as Density-Based Spatial Clustering of Applications with Noise (DBSCAN) and Support Vector Data Description (SVDD), fail to consider the influence

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNNLS.2025.3542719, IEEE Transactions on Neural Networks and Learning Systems

of past time points on future states, making it difficult to generalize these approaches to unseen real-world scenarios. Therefore, we propose using the fusion attention map as the criterion for anomaly detection, as it highlights the distinguishable association density between anomaly points and normal points with global and local sequences. Nevertheless, the attention map cannot be used to quantify landslide risk, as it presents the distribution of association weights rather than failure probabilities. To address this issue, we introduce the Gaussian kernel with a learnable parameter σ to reframe and quantify the fusion attention map.

B. Overview

As shown in Fig. 1, the encoder and decoder in the Fusionformer both consist of N identical modules, which include two key sub-modules: the multi-head FAM layer and the fully connected feed-forward network. To solve the problem of gradient vanishing, a residual connection is used on each of the two components, followed by layer normalization.

In the Fusionformer, the encoder first maps the input sequence $\mathcal{X}_{1:T}$ to latent variables, which are then fed to the decoder to predict future values $\hat{\mathcal{Y}}_{T+1:T+\tau}$ within a defined horizon. Additionally, to improve prediction performance, a discriminator is attached alongside the Fusionformer to learn the dynamic properties of real-world distributions.

C. Segment-Wise Sequence Embedding and Positional Encoding

Multiple adjacent points in the time domain can form an informative segment. As shown in Fig. 2, neighboring data points contain similar temporal patterns due to continuity. Therefore, we argue that segment-wise representation learning is more useful than point-wise representation learning.

In this part, we propose an SWSE method, which first partitions the input sequence of each variable into multiple non-overlapped segments of length L_{seg} and then embeds each segment into a learnable vector. The formulation of the SWSE is as follows:

$$\mathcal{X}_{1:T} = \left\{ \mathbf{x}_{i,d} \mid 1 \le i \le \frac{T}{L_{\text{seg}}}, 1 \le d \le D \right\}, \tag{2}$$

$$\mathbf{x}_{i,d} = \{ x_{t,d} \mid (i-1) \times L_{\text{seg}} \le t \le i \times L_{\text{seg}} \}, \tag{3}$$

where $\mathbf{x}_{i,d}$ is the i-th segment with length L_{seg} in dimension d, and $x_{t,d}$ is the t-th time step in the i-th segment of dimension d. Then, each segment is embedded into a vector by using learnable linear transformation with trainable positional encoding:

$$\mathbf{u}_{i,d} = W\mathbf{x}_{i,d} + W_{i,d}^{(\text{pos})}, \tag{4}$$

where $\mathbf{u}_{i,d}$ denotes embedded vector, and $W \in \mathbb{R}^{d_{\text{model}} \times L_{\text{seg}}}$ is the learnable projection matrix for segment embedding. $W_{i,d}^{(\text{pos})} \in \mathbb{R}^{d_{\text{model}}}$ is the learnable positional encoding for position (i,d), aiming to provide the sequential nature of temporal patterns. After that, a two-dimensional vector array characterizing the $\mathcal{X}_{1:T}$ can be obtained as follows:

$$\mathcal{U} = \{ \mathbf{u}_{i,d} \mid 1 \le i \le L_{\text{sn}}, 1 \le d \le D \}, \tag{5}$$

where $\mathbf{u}_{i,d}$ is a univariate time series segment of dimension d, and $L_{\text{sn}} = \frac{T}{L_{\text{seg}}}$ (sn is an abbreviation of segment number).

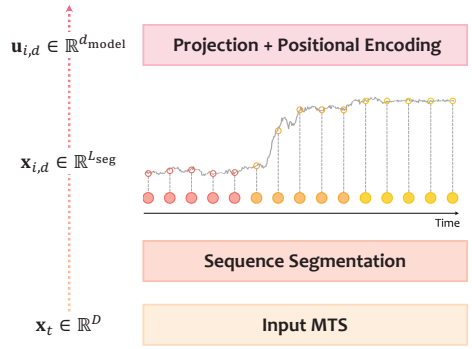


Fig. 2: Illustration of the SWSE.

D. Fusion Attention Mechanism

1) Segment-Wise Intra-Variable Attention: A longer look-back window is helpful in improving prediction capability because it allows the model to learn event patterns from the comprehensive context, as proven by [29]. However, simply prolonging the input length T comes at the cost of computational effort. To address this issue, the SWAA is proposed to learn reliable intra-variable dependencies from multiple short, low-complexity, and information-rich segments.

Given a 2-dimension array $\mathcal{U} \in \mathbb{R}^{D \times L_{\text{sn}} \times d_{\text{model}}}$ as the input of the SWAA, the dependencies between different segments in dimension d can be learned by:

$$\mathcal{Z}_{d,:}^{l,\text{tim}} = \text{LayerNorm}\left(\mathcal{U}_{d,:}^{l-1,\text{tim}} + \text{MSWAA}\left(\mathcal{Q}_A, \mathcal{K}_A, \mathcal{V}_A\right) \right), \tag{6}$$

$$\mathcal{U}^{l,\text{tim}} = \text{LayerNorm}\left(\mathcal{Z}^{l,\text{tim}} + \text{Feed-Forward}\left(\mathcal{Z}^{l,\text{tim}} \right) \right), \tag{7}$$

where $\mathcal{U}_{d,:}^{l-1,\text{tim}} \in \mathbb{R}^{L_{\text{sn}} \times d_{\text{model}}}$ represents all segments in dimension d of the $(l-1)$-th layer, $1 \le l \le L$ is the layer number of MSWAA. $1 \le d \le D$ is the dimension of MTS, $\mathcal{Z}^{l,\text{tim}} \in \mathbb{R}^{D \times L_{\text{sn}} \times d_{\text{model}}}$ is the hidden representation of the l-th layer, and LayerNorm is a widely used activation function in Transformer-based models [34]. Feed-Forward is the fully connected feed-forward network. MSWAA denotes a multi-head SWAA layer, where $\mathcal{Q}_A = W_{\mathcal{Q}}^l \mathcal{U}_{d,:}^{l-1,\text{tim}}$, $\mathcal{K}_A = W_{\mathcal{K}}^l \mathcal{U}_{d,:}^{l-1,\text{tim}}$, and $\mathcal{V}_A = W_{\mathcal{V}}^l \mathcal{U}_{d,:}^{l-1,\text{tim}}$ are used as query, key, and value for SWAA. $W_{\mathcal{Q}}^l$, $W_{\mathcal{K}}^l$, $W_{\mathcal{V}}^l \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$ are the parameter matrices of $\mathcal{Q}_A, \mathcal{K}_A, \mathcal{V}_A$ in the l-th layer respectively. The multi-head version of SWAA is illustrated in Fig. 3.

2) Segment-Wise Inter-Variable Attention: Inter-variable dependence is critical to the MTSF, i.e., for the target variable, exploiting the characteristics of related time series from other variables may help to improve prediction accuracy. Some previous works implicitly capture cross-variable dependency information from the latent feature space via CNN or GNN. Nevertheless, the above neural models consider the dependency weights (the degree of interaction between different variables varies, and this situation can be modeled by specific

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNNLS.2025.3542719, IEEE Transactions on Neural Networks and Learning Systems

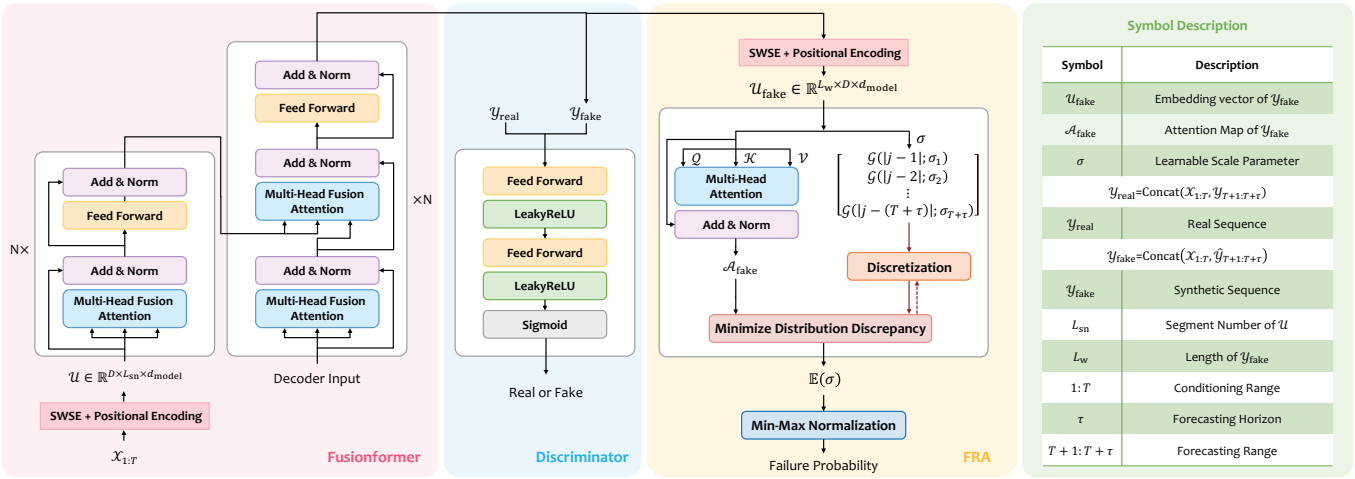


Fig. 1: Architecture of the Fusionformer model and the FRA method.

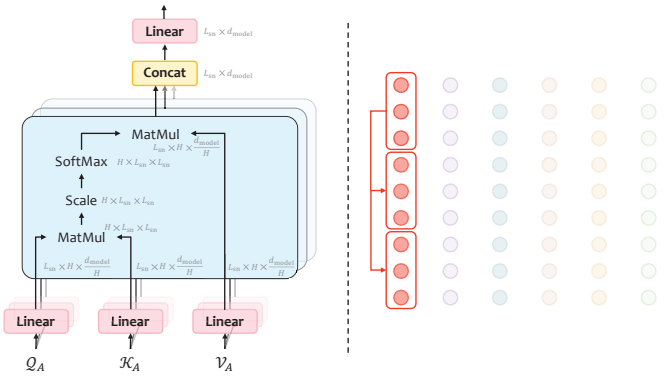


Fig. 3: Multi-head segment-wise intra-variable attention consists of h attention layers running in parallel.

methods as dependency weights) among different variables to be equal, which may degrade prediction performance. To overcome this limitation, the SWEA is developed in this paper to learn hierarchical inter-variable dependencies.

Note that all segments within $\mathcal{U}^{\text{tim}} \in \mathbb{R}^{D\times L_{\text{sn}}\times d_{\text{model}}}$ are merged in SWEA to enable the Fusionformer to learn inter-variable associations on a large scale. To achieve this goal, the output of the L-th layer in MSWAA is transmitted to the first layer in MSWEA:

$$\mathcal{Z}^{1,\text{var}} = \text{LayerNorm}\left(\mathcal{U}^{L,\text{tim}} + \text{MSWEA}\left(\mathcal{Q}_E, \mathcal{K}_E, \mathcal{V}_E\right)\right), \tag{8}$$

$$\mathcal{U}^{1,\text{var}} = \text{LayerNorm}\left(\mathcal{Z}^{1,\text{var}} + \text{Feed-Forward}\left(\mathcal{Z}^{1,\text{var}}\right)\right), \tag{9}$$

where $\mathcal{U}^{L,\text{tim}} \in \mathbb{R}^{D\times(L_{\text{sn}}\times d_{\text{model}})}$ represents the output of MSWAA layer. MSEWA denotes a multi-head SWEA layer, where $\mathcal{Q}_E = W_{\mathcal{Q}}^{1,\text{var}}\mathcal{U}^{L,\text{tim}}$, $\mathcal{K}_E = W_{\mathcal{K}}^{1,\text{var}}\mathcal{U}^{L,\text{tim}}$, and $\mathcal{V}_E = W_{\mathcal{V}}^{1,\text{var}}\mathcal{U}^{L,\text{tim}}$ denote the query, key, and value of the 1-th MSWEA layer, respectively. Furthermore, the variable dependencies of k-th layer can be captured by:

$$\mathcal{Z}^{k,\text{var}} = \text{LayerNorm}\left(\mathcal{U}^{k-1,\text{var}} + \text{MSWEA}\left(\mathcal{Q}_E, \mathcal{K}_E, \mathcal{V}_E\right)\right), \tag{10}$$

$$\mathcal{U}^{k,\text{var}} = \text{LayerNorm}\left(\mathcal{Z}^{k,\text{var}} + \text{Feed-Forward}\left(\mathcal{Z}^{k,\text{var}}\right)\right), \tag{11}$$

where $\mathcal{Q}_E = W_{\mathcal{Q}}^{k,\text{var}}\mathcal{U}^{k-1,\text{var}}$, $\mathcal{K}_E = W_{\mathcal{K}}^{k,\text{var}}\mathcal{U}^{k-1,\text{var}}$, and $\mathcal{V}_E = W_{\mathcal{V}}^{k,\text{var}}\mathcal{U}^{k-1,\text{var}}$. $1 \le k \le K$ is the layer number of MSWEA. The multi-head version of SWEA is illustrated in Fig. 4.

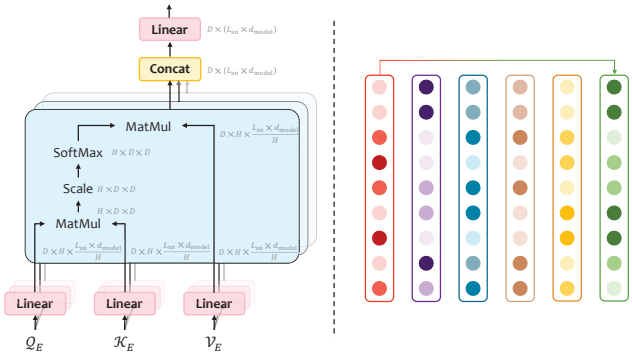


Fig. 4: Multi-head segment-wise inter-variable attention consists of h attention layers running in parallel.

E. Adversarial Learning

The adversarial learning approach used in this paper draws its inspiration from the generative adversarial network (GAN) [12], [36], [37]. Formally, let Θ_F and Θ_D represent the parameters of the Fusionformer F and the discriminator D, respectively. First, the Fusionformer is employed to generate future values close to the ground truth by optimizing the following function:

$$\Theta_F^* = \arg\min_{\Theta_F} L_F\left(\mathcal{Y}_{\text{fake}}; \Theta_F, \Theta_D^*\right), \tag{12}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNNLS.2025.3542719, IEEE Transactions on Neural Networks and Learning Systems

where Θ_F^* and Θ_D^* denote the optimal parameters of the Fusionformer and the discriminator, respectively. Furthermore, $\mathcal{Y}_{\text{fake}} \in \mathbb{R}^{L_{\text{w}} \times D}$ ($L_{\text{w}} = T + \tau$) is the synthetic sequence obtained by:

$$\mathcal{Y}_{\text{fake}} = \text{Concat}\left(\mathcal{X}_{1:T}, \hat{\mathcal{Y}}_{T+1:T+\tau}\right), \tag{13}$$

where Concat represents concatenation operation, and $\hat{\mathcal{Y}}_{T+1:T+\tau}$ is the predicted sequence. Moreover, L_F represents the loss function, which is formulated as follows:

$$L_F = \mathbb{E}\left[\log(1 - D(\mathcal{Y}_{\text{fake}}))\right], \tag{14}$$

where $\mathbb{E}$ is expectation, and $D(\cdot)$ denotes the output of the discriminator, which outputs 1 if the input is ground truth and 0 otherwise.

The discriminator in this paper is implemented with three fully connected layers and a Sigmoid activation function. Typically, the prediction loss is calculated using the ground truth and the predicted sequence within the future horizon τ. In contrast, the Fusionformer is optimized by reducing the global distribution discrepancy between the synthetic sequence $\mathcal{Y}_{\text{fake}}$ and real sequence $\mathcal{Y}_{\text{real}}$. On the one hand, calculating the global discrepancies can help the Fusionformer understand the overall trends and dynamic patterns of the MTS distribution. On the other hand, the discriminator can regularize the optimization direction of the Fusionformer in a global perspective, enhancing the prediction accuracy. Specifically, the discriminator is trained to distinguish the synthetic sequence from the real sequence by optimizing the following function:

$$\Theta_D^* = \arg\min_{\Theta_D} L_D\left(\mathcal{Y}_{\text{real}}, \mathcal{Y}_{\text{fake}}; \Theta_F^*, \Theta_D\right), \tag{15}$$

where $\mathcal{Y}_{\text{real}} \in \mathbb{R}^{L_{\text{w}} \times D}$ is the real sequence obtained by:

$$\mathcal{Y}_{\text{real}} = \text{Concat}\left(\mathcal{X}_{1:T}, \mathcal{Y}_{T+1:T+\tau}\right). \tag{16}$$

L_D represents the objective function of the real sequence and the synthetic sequence, which is formulated as follows:

$$L_D = -\mathbb{E}\left[\log(\mathcal{Y}_{\text{real}})\right] - \mathbb{E}\left[\log(1 - D(\mathcal{Y}_{\text{fake}}))\right]. \tag{17}$$

The training process of the proposed Fusionformer is shown in Algorithm 1.

Algorithm 1: The training process of the Fusionformer

Input: Historical series $\mathcal{X}_{1:T} = \{\mathbf{x}_1, \ldots, \mathbf{x}_T \mid \mathbf{x}_t \in \mathbb{R}^D\}$.
Input: Initial Fusionformer parameters Θ_F.
Output: Optimal parameters Θ_F^*.
Hyperparameters: L_{seg}, N_0, and N_1.
for $i_0 \leq N_0$ **do**
  for $i_1 \leq N_1$ **do**
    Predict $\hat{\mathcal{Y}}_{T+1:T+\tau}$ via Fusionformer;
    Construct $\mathcal{Y}_{\text{fake}}$ by concatenating $\mathcal{X}_{1:T}$ and $\hat{\mathcal{Y}}_{T+1:T+\tau}$;
    Update the discriminator D by descending stochastic gradient, which is formulated by: $\nabla_{\Theta_D} - \mathbb{E}\left[\log(\mathcal{Y}_{\text{real}})\right] - \mathbb{E}\left[\log(1 - D(\mathcal{Y}_{\text{fake}}))\right]$;
  end
  Update the Fusionformer F by descending stochastic gradient, which is formulated by: $\nabla_{\Theta_F} \mathbb{E}\left[\log(1 - D(\mathcal{Y}_{\text{fake}}))\right]$
end

F. Fusionformer-Based Risk Assessment

The requirement of SFEW is to assess the landslide risk quantitatively, i.e., to provide an exact probability value for the landslide occurrence, which is more challenging than the anomaly detection task. Generally, the SFEW method identifies the potential landslide by analyzing only the predicted slope state, however, the instability of real-world radar signals may affect the assessment accuracy. Therefore, we should extract the informative context from the combination of observed and predicted values to make an accurate risk assessment. In the Fusionformer, the self-attention mechanism is well used to learn the dynamic temporal patterns of a single variable and the interactions among all variables, which inspires us to exploit the attention map to characterize the movement trend of a slope. Nevertheless, the attention map cannot be used to quantify landslide risk, as it presents the distribution of association weights rather than failure probabilities. To address this issue, a Gaussian probability density function with a learnable parameter σ is introduced into the FRA method.

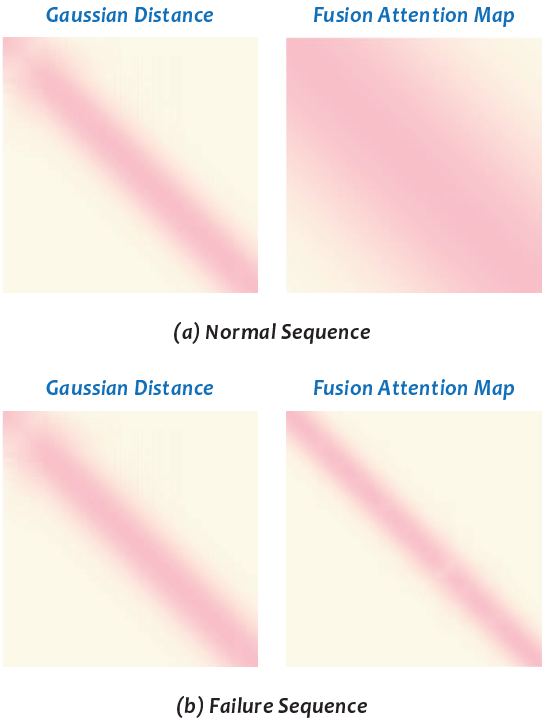


Fig. 5: Discrepancy between Gaussian distribution and fusion attention map.

Benefiting from the unimodal property of the Gaussian kernel [46], the discrete Gaussian distribution has similar characteristics to the attention map, as shown in Fig. 5. Specifically, for the failure sequence, anomalies can establish a robust association with adjacent time points, which are characterized by small values of σ. Furthermore, dominant normal time points can form informative associations with the

overall sequence, not limiting to the neighboring region, which is characterized by large values of σ. Thus, the parameter σ for quantifying the failure risk can be obtained by minimizing the discrepancy between the Gaussian distribution and the attention map. The implementation process of the FRA method is illustrated as follows.

First, for the i-th time point, its association weight to the j-th time point can be presented by Gaussian Kernel as:

$$\mathcal{G}_{\text{fake}}^{l} = \text{Rescale}\left(\frac{1}{\sqrt{2\pi}\sigma_i}\exp\left(-\frac{|j-i|^2}{2\sigma_i^2}\right)\right), \tag{18}$$

where $\mathcal{G}_{\text{fake}}^{l} \in \mathbb{R}^{L_{\text{w}} \times L_{\text{w}}}$ represents the distribution of the learned scale parameter $\sigma \in \mathbb{R}^{L_{\text{w}} \times 1}$, where the σ_i is associated with the i-th time point in $\mathcal{Y}_{\text{fake}}$. $\text{Rescale}\,(\cdot)$ represents the rescaling operation that converts Gaussian distances into discrete distributions $\mathcal{G}^l$ by dividing the sum of the rows. Then, the attention map is calculated by:

$$\mathcal{A}_{\text{fake}}^{l} = \text{Softmax}\left(\frac{\mathcal{Q}_F \left(\mathcal{K}_F\right)^T}{\sqrt{d_{\text{model}}}}\right), \tag{19}$$

where $\mathcal{A}_{\text{fake}}^{l} \in \mathbb{R}^{L_{\text{w}} \times L_{\text{w}}}$ is the attention map, and Softmax normalizes the attention map along the last dimension. $\mathcal{Q}_F = W_{\mathcal{Q}}^{l}\mathcal{Y}_{\text{fake}}^{l-1}$, $\mathcal{K}_F = W_{\mathcal{K}}^{l}\mathcal{Y}_{\text{fake}}^{l-1}$, $\mathcal{V}_F = W_{\mathcal{V}}^{l}\mathcal{Y}_{\text{fake}}^{l-1}$ are the query, key, and value in the l-th layer respectively. After that, the discrepancy L_{FRA} between the Gaussian distribution and the attention map is reduced by minimizing:

$$L_{\text{FRA}} = \frac{1}{L}\sum_{l=1}^{L} \text{KL}\left(\mathcal{A}_{\text{fake}}^{l} \parallel \mathcal{G}_{\text{fake}}^{l}\right), \tag{20}$$

where $\text{KL}\,(\cdot \parallel \cdot)$ denotes the Kullback-Leibler divergence used to calculate the difference between two discrete distributions corresponding to $\mathcal{A}_{\text{fake}}^{l}$ and $\mathcal{G}_{\text{fake}}^{l}$. Finally, the failure probability of i-th time point can be obtained by:

$$P_i = \frac{\sigma_{\text{n}} - \sigma_i}{\sigma_{\text{n}} - \sigma_{\text{f}}}, \tag{21}$$

where σ_{n} and σ_{f} are σ thresholds of normal and failure sequences chosen through extensive experiments.

V. Experimental Results on Slope Failure Datasets

According to most of the existing slope failure prediction models outlined in [15], slope displacement or velocity serve as the primary criteria for hazard assessment. Therefore, in this paper, we utilize displacement data as the target sequence for the SFEW task. Furthermore, the most reliable variables for state prediction are velocity and acceleration, as they directly correspond to the stability conditions of the moving mass. Moreover, we explicitly consider various weather data variables (such as humidity, temperature, pressure, accumulated rainfall, and wind speed) due to their significant influence on slope stability.

A. Datasets

This work is supported by the EU-funded DIG_IT project, which aims to enhance the efficiency and sustainability of mining operations through the development of a smart industrial IoT platform. All datasets used in this work are collected by the Titania radar software from an open-pit mine located in southern Norway. Fig. 6 shows a screenshot of the digital terrain model (DTM), illustrating the displacement recorded within a 24-hour period in one area of the mine. The individual pixel data within the blue circle is representative of measurements taken at a single point. Similarly, the same types of variables (displacement, velocity, acceleration, amplitude) are recorded for all pixels in the measured area.

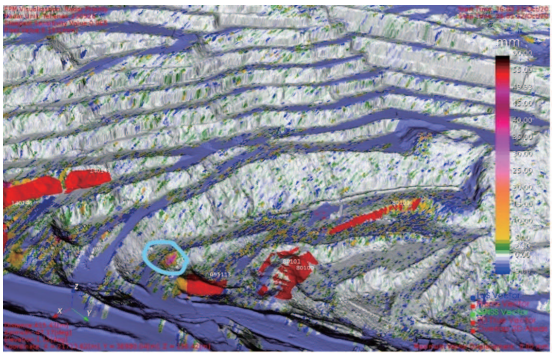


Fig. 6: DTM supplied by Titania AS.

In this section, we conduct experiments on the following four real-world datasets, including one normal dataset and three failure datasets.

- **OPMSmN** (Open-Pit Mine Slope-minutely Normal) contains 9999 normal slope records collected between 27 January 2021 and 24 February 2021.
- **OPMSmF1** (Open-Pit Mine Slope-minutely Failure) contains 3059 landslide records collected between 24 December 2018 and 1 January 2019.
- **OPMSmF2** contains 2339 landslide records collected between 7 July 2022 and 13 July 2022.
- **OPMSmF3** contains 1064 landslide records collected between 19 March 2023 and 21 March 2023.

Each slope data includes eight attributes/variables: velocity, acceleration, humidity, temperature, pressure, cumulative rainfall, wind speed, and displacement. Data points in each dataset are recorded every 4 minutes. The split ratio of the training/validation/test set is $0.7 : 0.1 : 0.2$.

B. Baselines

A comprehensive set of eight methods, including four categories, is used to test the Fusionformer.

- **LSTM** [35] controls the flow of information by using special multiplicative units called gates.
- **TCN** [3] attempts to model the temporal pattern using dilated causal convolutions.
- **LSTMa** [2] encodes the input MTS into a multi-dimensional vector and automatically aligns each target

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNNLS.2025.3542719, IEEE Transactions on Neural Networks and Learning Systems

TABLE I: MAE and MSE at different forecasting lengths. Bold denotes the best results. Yellow background marks the LSTM and TCN models; purple marks the LSTM-based models with the attention mechanism; green marks the Transformer-based models; red marks the Fusionformer proposed in this paper.

Models		LSTM		TCN		LSTMa		LSTnet		Transformer		Informer		FEDformer		Crossformer		Fusionformer	
Metric		MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
OPMSmN	16	0.603	0.439	0.623	0.443	0.429	0.236	0.369	0.157	0.251	0.428	0.289	0.196	0.184	**0.129**	0.350	0.147	**0.180**	0.186
	32	0.626	0.460	0.630	0.514	0.443	0.259	0.472	0.240	0.288	0.365	0.304	0.210	0.196	0.139	0.352	0.165	**0.183**	**0.132**
	64	0.639	0.450	0.601	0.412	0.500	0.310	0.476	0.255	0.332	0.366	0.313	0.219	0.214	0.155	0.369	0.163	**0.153**	**0.118**
	128	0.588	0.421	0.647	0.482	0.523	0.342	0.576	0.345	0.545	0.846	0.383	0.294	0.255	0.199	0.434	0.211	**0.250**	**0.195**
	256	0.682	0.524	0.689	0.541	0.603	0.427	0.575	0.337	0.531	0.537	0.413	0.331	**0.287**	0.242	0.429	**0.207**	0.338	0.247
OPMSmF1	16	1.037	1.788	0.897	1.427	0.490	0.351	0.599	0.416	0.469	0.319	1.449	3.795	0.438	0.327	0.435	0.231	**0.225**	**0.114**
	32	1.136	2.169	0.305	0.337	0.499	0.353	0.634	0.458	0.413	0.245	1.538	4.729	0.323	**0.146**	0.476	0.275	**0.291**	0.194
	64	1.203	2.811	0.854	1.282	0.501	0.342	0.630	0.462	0.431	0.263	1.632	5.070	0.437	0.299	0.537	0.296	0.325	**0.248**
	128	1.401	4.259	0.929	1.390	0.477	0.323	0.638	0.485	0.433	0.274	1.965	7.849	0.628	0.542	0.576	0.345	**0.337**	**0.238**
	256	2.090	9.601	0.859	1.165	0.565	0.416	0.693	0.574	0.468	0.318	1.684	6.640	0.632	0.547	0.456	**0.241**	**0.344**	0.289
OPMSmF2	16	0.551	0.566	0.518	0.274	0.532	0.363	0.385	0.228	0.299	0.309	0.658	0.791	0.349	0.269	0.350	0.171	**0.228**	0.123
	32	0.664	0.854	0.551	0.308	0.617	0.458	0.541	0.363	0.330	0.344	0.719	0.943	0.550	0.471	0.599	0.426	**0.258**	0.282
	64	0.638	1.020	0.568	0.329	0.631	0.470	0.636	0.475	0.457	0.511	0.791	1.244	0.515	0.406	0.550	0.367	**0.307**	**0.211**
	128	0.713	1.163	0.520	0.280	0.665	0.516	0.644	0.499	0.573	0.641	0.806	1.145	0.485	0.390	0.549	0.353	**0.344**	**0.276**
OPMSmF3	16	0.575	0.404	0.577	0.398	0.356	**0.175**	0.503	0.274	0.252	0.308	2.272	14.081	1.224	3.662	0.464	0.227	**0.239**	0.290
	32	0.455	0.299	0.592	0.434	0.482	0.308	0.528	**0.293**	**0.287**	0.348	2.665	14.838	1.316	3.852	0.551	0.308	0.295	0.347
	64	0.526	0.367	0.608	0.434	0.540	0.355	0.542	**0.300**	0.419	0.528	2.400	14.848	1.701	6.976	0.568	0.329	**0.379**	0.457
Count		0		0		1		2		1		0		3		2		22	

value to the relevant past point based on the encoder-decoder RNN.

- **LSTnet** [19] employs CNN to capture short-term local dependencies and utilizes RNN to extract long-term trends in the data over time.
- **Transformer** [34] captures cross-time dependencies by stacking multi-head self-attention layers and feed-forward layers.
- **Informer** [59] is a Transformer-based forecasting model employing the ProbSparse self-attention to capture cross-time dependencies.
- **FEDformer** [60] is a Transformer-based forecasting model that uses seasonal trend decomposition with frequency-enhanced attention blocks to capture cross-time dependencies.
- **Crossformer** [56] is a Transformer-based MTSF model that uses the two-stage attention layer to capture the cross-time and cross-dimension dependencies.

C. Evaluation Metrics

MSE and MAE are employed as metrics to evaluate the performance of the Fusionformer, and they are defined as follows:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2, \tag{22}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right|, \tag{23}$$

where n represents the number of samples, y and $\hat{y}$ denote the ground truth and predicted signal, respectively. Additionally, recall, specificity, and accuracy are used to measure the performance of the FRA method, and they are formulated as follows:

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}, \tag{24}$$

$$\mathrm{Specificity} = \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}}, \tag{25}$$

$$\mathrm{Accuracy} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FN}+\mathrm{FP}+\mathrm{TN}}, \tag{26}$$

where TP is the correctly identified failure, FP is the incorrectly identified failure, TN is the correctly monitored normal state, and FN is the incorrectly monitored normal state.

D. Implementation Details

We follow the experimental setup of [59], i.e., the training set, validation set, and test set are zero-mean normalized by the mean and standard deviation of the training set. We roll the whole sequence with $\mathrm{stride} = 1$ to produce different input and output pairs. We assess the performance of all models over the changing future horizon τ. All experiments are repeated five times and the average of each metric is reported.

The Fusionformer contains 4 encoder layers and 3 decoder layers. The dimension of hidden state is set to 256. The head number of FAM layer is set to 6. The segment length is set to 32. Adam optimizer is employed for model training and the learning rate is selected from $\{5\mathrm{e}-3, 1\mathrm{e}-3, 5\mathrm{e}-4, 1\mathrm{e}-4, 5\mathrm{e}-5, 1\mathrm{e}-5\}$ through grid search. The batch size is set to 32, and the total number of epochs is set to 30 with a proper early stopping strategy.

For different datasets, the τ is progressively prolonged, i.e., $\{16\mathrm{min}, 32\mathrm{min}, 64\mathrm{min}, 128\mathrm{min}, 256\mathrm{min}\}$. Note that the Fusionformer can predict up to 128 minutes on OPMSmF2 and up to 64 minutes on OPMSmF3 due to insufficient data. Moreover, in sensitivity analysis, we treat T as a hyperparameter and test the impact of different input lengths on prediction performance.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNNLS.2025.3542719, IEEE Transactions on Neural Networks and Learning Systems

For baselines, all hyperparameters (except T) recommended in the original paper are adopted if the dataset used in the original paper is the same as ours. Otherwise, the hyperparameters are selected via the grid search method using the validation set. All models, including the Fusionformer and baselines, are implemented by PyTorch and trained on a single NVIDIA GEFORCE RTX 3090 GPU with 24GB memory.

E. Forecasting Performance Evaluation

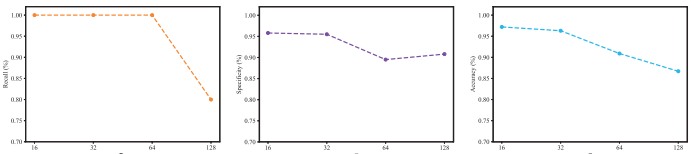
Table I summarizes the evaluation results of the Fusionformer and all baselines on the four datasets, where a progressively longer prediction horizon is used to measure model performance. From Table I, we can draw the following conclusions:

- Our model (Fusionformer) shows leading performance on all four datasets (see the counts in the last row). Different from the baselines, the prediction error of the Fusionformer rises smoothly with increasing prediction horizon, which indicates the success of the Fusionformer in improving the precision of MTS forecasting.
- Deep learning methods (LSTM and TCN) do not explicitly model the dependencies between different variables. As a result, they perform worse than LSTMa, LSTnet, Transformer-based methods, and Fusionformer on most datasets. Specifically, the Fusionformer algorithm achieves significantly better results than the RNN-based LSTM and the CNN-based TCN. For the input-96-prediction-128 setting, compared to TCN, the Fusionformer decreases MSE by **59.5%** in OPMSmN and **82.8%** in OPMSmF1.
- LSTMa and LSTnet achieve slightly better performance than LSTM and TCN, which is attributed to the use of the attention mechanism. Nevertheless, their results are still not comparable to those of the Fusionformer in terms of MAE and MSE, i.e., 22 >3. This demonstrates that the self-attention mechanism is more suitable for learning long-term temporal patterns than the recursive structure. Note that for the OPMSmF3 dataset, LSTMa achieves the best performance at prediction horizon 16, and LSTnet performs best at horizons 32 and 64. One possible explanation for this phenomenon is that the OPMSmF3 dataset has only 1064 records, leading to the underfitting problem of Transformer-based algorithms, which have much more network parameters than LSTMa and LSTnet.
- Benefiting from the self-attention mechanism, Transformer, Informer, FEDformer, and Crossformer significantly surpass deep learning methods, where the FEDformer algorithm performs better than the other Transformer-based methods. However, the proposed Fusionformer still shows a significant advantage over FEDformer in terms of MSE, with average reductions of **83.7%** (at 16), **79.3%** (at 32), and **78.1%** (at 64). This suggests that the cooperation of the fusion attention mechanism and adversarial learning is more effective than the conventional attention mechanism in capturing intra- and inter-variable associations in the MTS.

Fig. 7 displays the forecasting cases for three variables in the OPMSmF2 dataset with prediction length $\tau = 128$. For the variable "Acceleration", all seven models can learn dynamic characteristics from the sequence, but the Fusionformer is the closest to the ground truth. For "Velocity", LSTnet and Transformer only predict a rough trend, in addition Informer, FEDformer, Crossformer capture the non-deterministic dynamic variations to a certain extent, while the Fusionformer recovers all feature details and achieves the best prediction performance. For "Displacement", all models capture the ascending trend of the sequence, and the curve predicted by the Fusionformer is sharper than the other six models.

F. Detection Performance Evaluation

Fig. 8 shows the recall, specificity, and accuracy of the risk assessment. It can be found that the FRA has a recall of 100%, a specificity of 96%, and an accuracy of 97% when $\tau = 16$, which is an outstanding experimental result. Note that low recall represents a high missing alarm rate and low specificity represents a high false alarm rate. As shown in Fig. 8(a), the FRA achieves extremely low missing alarm rates with the varying τ from 16 to 64, which means that the FRA can learn a low σ value to represent the failure time points. Furthermore, some prominent event patterns in the normal data may be misidentified as failures due to the unimodal property of the Gaussian distribution, which results in the increasing false alarm rate within an acceptable range and this issue will be addressed in future work. Additionally, although these three metrics decrease to some extent with the extension of the prediction horizon, they still meet the engineering requirements.



(a) Recall (b) Specificity (c) Accuracy

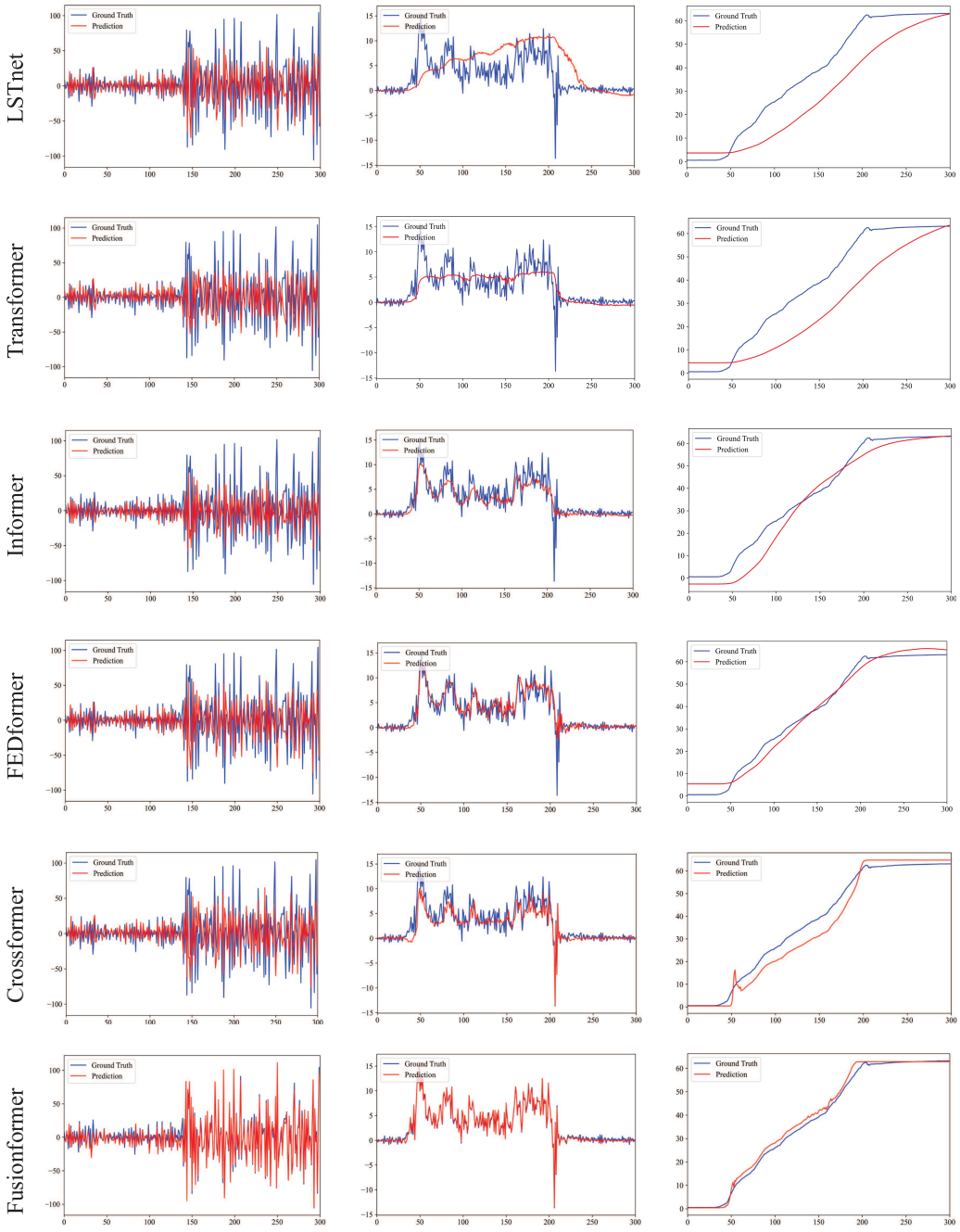
Fig. 8: Recall, specificity, and accuracy of slope failure risk assessment results.

G. Hyper-Parameter Sensitivity Analysis

The segment length L_{seg} is first determined, as it is directly related to the forecasting performance and the computational complexity of the Fusionformer.

As shown in Fig. 9(a), a short L_{seg} fails to capture long-term temporal patterns, while a long L_{seg} is difficult to be learned due to the limited computational ability, and therefore the best results are obtained when $L_{\text{seg}} = 32$. Fig. 9(b) shows the MSE and MAE of the Fusionformer with the number of input lengths T ranging from 48 to 672, where the empty column at 672 indicates an out-of-memory problem. The best results are obtained when T is set to 192. The reason for this phenomenon may be that a small T provides limited information about past time points, while a large T would introduce noise. Fig. 9(c) shows the prediction performance of the Fusionformer with the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNNLS.2025.3542719, IEEE Transactions on Neural Networks and Learning Systems



(a) Acceleration (b) Velocity (c) Displacement

Fig. 7: Showcases of the three dimensions (acceleration, velocity, and displacement) in the OPMSmF2 dataset with prediction length $\tau = 128$. The red and blue curves indicate prediction and ground truth, respectively. Each row represents a model and each column represents a variable.

growing number of encoder layers from 1 to 5. Generally, the Fusionformer with a large number of encoder layers can learn more accurate information than a small number of encoder layers, but it also requires numerous computational resources. Finally, the head number of FAM can be determined based on the available computational resources, as the results of the MSE and MAE are stable.

VI. Experimental Results on Public MTSF Datasets

In this section, the performance of the proposed Fusionformer is comprehensively evaluated on various public time series forecasting datasets.

A. Datasets

We extensively employ five MTSF datasets in the experiments, including:

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNNLS.2025.3542719, IEEE Transactions on Neural Networks and Learning Systems

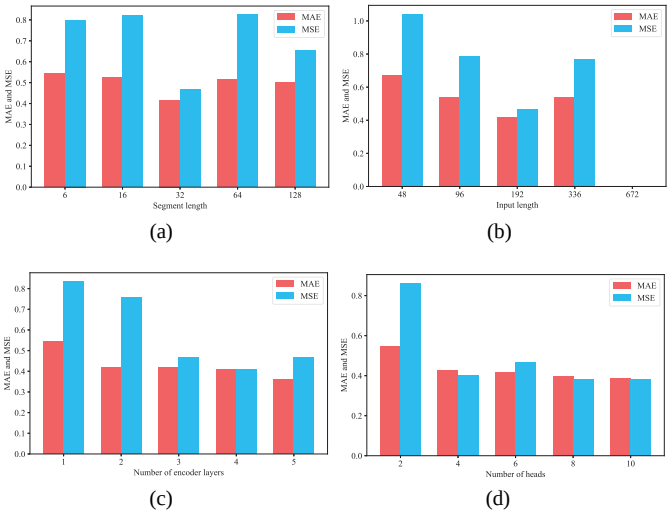


Fig. 9: Impacts of key parameters on forecasting performance.

- **ETT** dataset [59] which consists of four subsets, including two hourly datasets (ETTh1 and ETTh2) and two 15-minute datasets (ETTm1 and ETTm2). Each subset contains seven factors related to electricity transformers, covering the period from July 2016 to July 2018.
- **Exchange** dataset [19] collects daily exchange rate data for eight countries from 1990 to 2016.
- **Traffic** dataset [60] includes hourly roadway occupancy rates recorded by San Francisco freeway sensors from 2015 to 2016.
- **Weather** dataset comprises 21 meteorological indicators in Germany over one year, including temperature, humidity, etc.
- **Electricity Consuming Load (ECL)** dataset [44] records the hourly electricity consumption of 321 customers from 2012 to 2014.

Detailed descriptions of these datasets are provided in Table II.

TABLE II: Detailed descriptions of the datasets.

Dataset	Dim	Train/Validation/Test Size	Prediction Length
ETTh1	7	(8545, 2881, 2881)	96, 192, 336, 720
ETTh2	7	(8545, 2881, 2881)	96, 192, 336, 721
ETTm1	7	(34465, 11521, 11521)	96, 192, 336, 722
ETTm2	7	(34465, 11521, 11522)	96, 192, 336, 723
Exchange	8	(5120, 665, 1422)	96, 192, 336, 724
Traffic	862	(12185, 1757, 3509)	96, 192, 336, 725
Weather	21	(36792, 5271, 10540)	96, 192, 336, 726
ECL	321	(18317, 2633, 5261)	96, 192, 336, 727

B. Baselines

We employ six well-acknowledged forecasting models used in the iTransformer [24] as our benchmarks, including:

- **TCN-based methods**: SCINet [23] employs a recursive downsample-convolve-interact architecture for temporal modeling and forecasting; TimesNet [43] models temporal variation by a modular architecture and captures intraperiod- and interperiod-variations in 2D space by a parameter-efficient inception block.
- **Transformer-based methods**: Autoformer [44] is a variant of Transformer that builds on the decomposition architecture with an autocorrelation mechanism; Stationary [25] employs two interdependent modules, Series Stationarization, and De-stationary Attention, for non-stationarity real-world data prediction.
- **Linear-based methods**: DLinear [53] decomposes a raw data input into a trend component by a moving average kernel and forecasts future time series with two one-layer linear layers; TiDE [6] is an encoder-decoder model based on a multi-layer perceptron.

C. Ablation Study

To verify the effectiveness of the Fusionformer components, we conduct detailed ablations on the ETTh1, Traffic, and Weather datasets in line with Crossformer [56]. We use the vanilla Transformer as the baseline and **SWSE+FAM+AL** denote the Fusionformer without ablation. Three ablation variations are compared: (1) SWSE; (2) SWSE+FAM; (3) SWSE+AL. Results are averages of all prediction lengths including 24, 168, and 720. The corresponding results are recorded in Table III, where the best result is highlighted as **red** and the sub-optimal result is labeled with **blue**.

TABLE III: Component ablation of Fusionformer.

Ablation Variations	ETTh1		Traffic		Weather	
	MSE	MAE	MSE	MAE	MSE	MAE
Transformer	0.782	0.936	0.502	0.622	0.575	0.638
SWSE	0.644	0.734	0.458	**0.490**	0.463	0.450
SWSE+FAM	**0.418**	**0.398**	**0.356**	**0.487**	0.459	0.425
SWSE+AL	0.472	0.432	0.404	0.518	**0.418**	**0.388**
SWSE+FAM+AL	**0.321**	**0.272**	**0.307**	0.495	**0.285**	**0.231**

Detailed analysis are provided: (1) The only distinction between the vanilla Transformer and SWSE is the embedding method. SWSE aggregates time steps into semantically rich sub-sequence patches, achieving an average 50.88% promotion on vanilla Transformer. (2) FAM effectively improves the prediction precision on top of SWSE, which means it can capture multivariate semantic correlations rather than difficult and inflexible temporal correlations compared to the standard self-attention mechanism. (3) The advantage of adversarial learning lies in better capturing the complex characteristics of data distribution and predicting more realistic sequences from the global receptive field, compared to traditional MSE, which tends to overemphasize point-to-point errors and results in blurry and less diverse outputs. (4) The best MTSF performance can only be achieved when all three components in the proposed Fusionformer algorithm work together. In other words, each component plays a significant and indispensable role in the Fusionformer algorithm.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNNLS.2025.3542719, IEEE Transactions on Neural Networks and Learning Systems

TABLE IV: MTSF results with prediction length $\tau \in \{96, 192, 336, 720\}$ and fixed lookback window $T = 96$. Results are averaged over all predicted lengths. *Average* denotes further averaged results over all subsets of the ETT.

Models	Fusionformer		SCINet		TimesNet		Autoformer		Stationary		Dlinear		TiDE	
Metric	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
ETT(*Average*)	0.448	**0.427**	0.597	0.689	**0.404**	**0.391**	0.459	0.465	0.464	0.471	**0.444**	0.442	0.470	0.482
Exchange	0.422	0.408	0.626	0.750	0.443	0.416	0.539	0.613	0.454	0.461	**0.414**	**0.354**	**0.413**	**0.370**
Traffic	**0.289**	**0.380**	0.509	0.804	0.336	0.620	0.379	0.628	0.340	0.624	0.383	0.625	0.473	0.760
Weather	**0.313**	**0.254**	0.363	0.292	**0.287**	**0.259**	0.382	0.338	0.314	0.288	0.317	0.265	0.320	0.271
ECL	**0.284**	**0.200**	0.365	0.268	**0.295**	**0.192**	0.338	0.227	0.296	0.193	0.300	0.212	0.344	0.251
Count	7		0		6		0		0		3		2	

D. Main Results

Comprehensive results are recorded in Table IV, where the best result is highlighted as **red** and the sub-optimal result is labeled with **blue**. Compared with other models, Fusionformer shows superior performance on most datasets, as well as on different prediction lengths, with the 4 top-1 and 3 top-2 metrics out of 10 in total. It is worth noting that Fusionformer is particularly adept at high-dimensional forecasting tasks, benefiting from the transformation of the token embedding paradigm from time-centric to variate-centric.

VII. Experimental Results on Anomaly Detection Datasets

In this section, we comprehensively evaluate the performance of the proposed FRA on various public anomaly detection datasets.

A. Datasets

We extensively employ five anomaly detection datasets used by [46], including: (1) **SMD** contains service monitoring data in five weeks from 28 different machines. (2) **SMAP and MSL** dataset contains real spacecraft telemetry data and anomalies from the Soil Moisture Active Passive satellite and the Curiosity Rover on Mars. (3) **SWaT** contains 7 days of normal operation data and 4 days of attack scenario data collected by 51 sensors from a real-world industrial water treatment plant. Detailed descriptions of these datasets are provided in Table V.

TABLE V: Detailed descriptions of the anomaly detection datasets.

Dataset	Dim	Train/Validation/Test Size	Prediction Length
SMD	38	(566724, 141681, 708420)	100
MSL	55	(44653, 11664, 73729)	100
SMAP	25	(108146, 27037, 427617)	100
SWaT	51	(396000, 99000, 449919)	100

B. Baselines

We add three well-acknowledged anomaly detection models as our benchmarks, including: (1) Anomaly Transformer [46] proposes an Anomaly-Attention mechanism for amplifying the normal and abnormal distinguishability of the association discrepancy for anomaly detection; (2) MAD-GANs [20] is an unsupervised multivariate anomaly detection method that proposes an anomaly score called DR-score to detect anomalies through discrimination and reconstruction; and (3) TadGAN [11] is trained with a cycle consistency loss to efficiently reconstruct time series data.

C. Main Results

Anomaly detection performance of the FRA and three comparison models are evaluated on various datasets using Precision, Recall, and F1-Score metrics. The corresponding results are recorded in Table VI, where the best result is highlighted as **red** and the sub-optimal result is labeled with **blue**. It can be found that FRA performs well in anomaly detection, outperforming advanced GAN-based models and having competitive capabilities with Anomaly Transformer. Specifically, neither MAD-GAN, which purely relies on adversarial learning to model time series distributions, nor TadGAN and Anomaly Transformer, which employs a vanilla attention mechanism to capture point-to-point correlations, achieves results as favorable as FRA. This is primarily due to their inability to effectively handle complex long-term dependencies and broader multivariate correlations.

TABLE VI: Anomaly detection task. A higher value of Precision, Recall, F1-Score indicates a better performance. * represents Transformer.

Models		FRA	MAD-GAN	TadGAN	Anomaly*
SMD	P	88.95	**99.91**	**92.62**	89.40
	R	**98.93**	84.40	**99.74**	95.45
	F1	**93.67**	91.50	**96.05**	92.23
MSL	P	**93.12**	85.16	90.38	**92.09**
	R	98.92	**99.30**	**99.99**	95.15
	F1	**95.93**	91.69	**94.94**	93.59
SMAP	P	**95.72**	81.57	80.43	**94.13**
	R	**100.00**	92.16	**99.99**	99.40
	F1	**97.81**	86.54	89.15	**96.69**
SWaT	P	**99.91**	95.93	**97.60**	91.55
	R	**97.27**	69.57	69.97	**96.73**
	F1	**98.57**	80.65	81.51	**94.07**

VIII. Conclusions

In this article, a Fusionformer algorithm has been proposed for MTSF. Specifically, we have introduced the SWSE method, which aggregates multiple adjacent points into an informative segment, enabling the model to learn meaningful temporal properties from historical data. Further, the FAM has been designed to learn long-term temporal dependencies and complex inter-variable associations within the MTS. We have also employed an adversarial learning approach to enhance the prediction accuracy of the Fusionformer with the aid of an auxiliary discriminator. The issue of SFEW in open-pit mining operations serves as a testbed to evaluate the performance of the proposed Fusionformer and FRA. Our experimental results attest to the effectiveness of the Fusionformer in predicting future time points and the capability of the FRA in assessing slope failure. In future work, we will focus on developing advanced fusion strategies [10], [18], [39], [41], [50]–[52] so as to further improve the efficiency and accuracy of our forecasting model.

References

[1] S. Asadi, A. Tavakoli, and S. R. Hejazi, A new hybrid for improvement of auto-regressive integrated moving average models applying particle swarm optimization, *Expert Systems with Applications*, vol. 39, no. 5, pp. 5332–5337, 2012.

[2] D. Bahdanau, K. Cho, and Y. Bengio, Neural machine translation by jointly learning to align and translate, In: *International Conference on Learning Representations (ICLR)*, 2015.

[3] S. Bai, J. Z. Kolter, and V. Koltun, An empirical evaluation of generic convolutional and recurrent networks for sequence modeling, *arXiv preprint arXiv:1803.01271*, 2018.

[4] L. Chen, D. Chen, Z. Shang, B. Wu, C. Zheng, B. Wen, and W. Zhang, Multi-scale adaptive graph neural network for multivariate time series forecasting, *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 10, pp. 10748–10761, 2023.

[5] N. Chen, S. Watanabe, J. Villalba, P. Żelasko, and N. Dehak, Non-autoregressive transformer for speech recognition, *IEEE Signal Processing Letters*, vol. 28, pp. 121–125, 2020.

[6] A. Das, W. Kong, A. Leach, S. Mathur, R. Sen, and R. Yu, Long-term forecasting with tide: Time-series dense encoder, *arXiv preprint arXiv:2304.08424*, 2023.

[7] S. G. Du, C. Saroglou, Y. Chen, H. Lin, and R. Yong, A new approach for evaluation of slope stability in large open-pit mines: A case study at the Dexing Copper Mine, China, *Environmental Earth Sciences*, vol. 81, no. 3, art. no. 102, 2022.

[8] H. Duan, Y. Li, H. Jiang, Q. Li, W. Jiang, Y. Tian, and J. Zhang, Retrospective monitoring of slope failure event of tailings dam using InSAR time-series observations, *Natural Hazards*, vol. 117, no. 3, pp. 2375–2391, 2023.

[9] S. Fu, S. Zhong, L. Lin, and M. Zhao, A novel time-series memory auto-encoder with sequentially updated reconstructions for remaining useful life prediction, *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 12, pp. 7114–7125, 2022.

[10] H. Gao, Y. Li, L. Yu and H. Yu, Collaborative-prediction-based recursive filtering for nonlinear systems with sensor saturation under duty cycle scheduling, *Systems Science & Control Engineering*, vol. 11, no. 1, art. no. 2247007, 2023.

[11] A. Geiger, D. Liu, S. Alnegheimish, A. Cuesta-Infante, and K. Veeramachaneni, Tadgan: Time series anomaly detection using generative adversarial networks, In: *2020 IEEE International Conference on Big Data*, 2020, pp. 33–43.

[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, Generative adversarial nets, In: *Advances in Neural Information Processing Systems (NIPS)*, 2014, pp. 2672–2680.

[13] F. Han, J. Liu, J. Li, J. Song, M. Wang, and Y. Zhang, Consensus control for multi-rate multi-agent systems with fading measurements: the dynamic event-triggered case, *Systems Science & Control Engineering*, vol. 11, no. 1, art. no. 2158959, 2023.

[14] J. Hu and W. Zheng, A deep learning model to effectively capture mutation information in multivariate time series prediction, *Knowledge-Based Systems*, vol. 203, art. no. 106139, Sep. 2020.

[15] E. Intrieri, T. Carlà, and G. Gigli, Forecasting the time of failure of landslides at slope-scale: A literature review, *Earth-Science Reviews*, vol. 193, pp. 333–349, 2019.

[16] M. Jin, Y. Zheng, Y.-F. Li, S. Chen, B. Yang, and S. Pan, Multivariate time series forecasting with dynamic graph neural ODEs, *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 9, pp. 9168–9180, 2023.

[17] N. Jin, Y. Zeng, K. Yan, and Z. Ji, Multivariate air quality forecasting with nested long short term memory neural network, *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 8514–8522, 2021.

[18] Y. Jin, X. Ma, X. Meng, and Y. Chen, Distributed fusion filtering for cyber-physical systems under Round-Robin protocol: a mixed H_2/H_∞ framework, *International Journal of Systems Science*, vol. 54, no. 8, pp. 1661–1675, 2023.

[19] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, Modeling long- and short-term temporal patterns with deep neural networks, In: *International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR)*, 2018, pp. 95–104.

[20] D. Li, D. Chen, B. Jin, L. Shi, J. Goh, and S. K. Ng, MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks, In: *Artificial Neural Networks and Machine Learning*, 2019, pp. 703–716.

[21] G. Li and J. J. Jung, Deep learning for anomaly detection in multivariate time series: Approaches, applications, and challenges, *Information Fusion*, vol. 91, pp. 93–102, Mar. 2023.

[22] X. Li, M. Li, P. Yan, G. Li, Y. Jiang, H. Luo, and S. Yin, Deep learning attention mechanism in medical image analysis: Basics and beyonds, *International Journal of Network Dynamics and Intelligence*, vol. 2, no. 1, pp. 93–116, Mar. 2023.

[23] M. Liu, A. Zeng, M. Chen, Z. Xu, Q. Lai, L. Ma, and Q. Xu, Scinet: Time series modeling and forecasting with sample convolution and interaction, In: *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 5816–5828.

[24] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long, iTransformer: Inverted Transformers are effective for time series forecasting, *arXiv preprint arXiv:2310.06625*, 2023.

[25] Y. Liu, H. Wu, J. Wang, and M. Long, Non-stationary transformers: Rethinking the stationarity in time series forecasting, In: *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 9881–9893.

[26] X. Luo, Y. Zhou, Z. Liu, and M. C. Zhou, Fast and accurate non-negative latent factor analysis on high-dimensional and sparse matrices in recommender systems, *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 4, pp. 3897–3911, Apr. 2023.

[27] G. Ma, S. Xu, B. Jiang, C. Cheng, X. Yang, Y. Shen, T. Yang, Y. Huang, H. Ding, and Y. Yuan, Real-time personalized health status prediction of lithium-ion batteries using deep transfer learning, *Energy & Environmental Science*, vol. 15, no. 10, pp. 4083–4094, 2022.

[28] W. T. Ng, K. Siu, A. C. Cheung, and M. K. Ng, Expressing multivariate time series as graphs with time series attention transformer, In: *International Joint Conference on Artificial Intelligence (IJCAI)*, 2022.

[29] Y. Nie, N. Nguyen, P. Sinthong, and J. Kalagnanam, A time series is worth 64 words: Long-term forecasting with Transformers, In: *International Conference on Learning Representations (ICLR)*, 2023.

[30] J. Qin, S. Du, J. Ye, and R. Yong, SVNN-ANFIS approach for stability evaluation of open-pit mine slopes, *Expert Systems with Applications*, vol. 198, art. no. 116816, 2022.

[31] R. R. Sharma, M. Kumar, S. Maheshwari, and K. P. Ray, EVDHM-ARIMA-based time series forecasting model and its application for COVID-19 cases, *IEEE Transactions on Instrumentation and Measurement*, vol. 70, art. no. 6502210, 2020.

[32] L. Tian, Multi-dimensional adaptive learning rate gradient descent optimization algorithm for network training in magneto-optical defect detection, *International Journal of Network Dynamics and Intelligence*, vol. 3, no. 3, art. no. 100016, Sept. 2024.

[33] A. Ulhaq, N. Akhtar, G. Pogrebna, and A Mian, Vision transformers for action recognition: A survey, *arXiv preprint arXiv:2209.05700*, 2022.

[34] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, Attention is all your need, In: *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[35] C. Wang, Z. Wang, W. Liu, Y. Shen, and H. Dong, A novel deep offline-to-online transfer learning framework for pipeline leakage detection with small samples, *IEEE Transactions on Instrumentation and Measurement*, vol. 72, art. no. 3503913, 2022.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNNLS.2025.3542719, IEEE Transactions on Neural Networks and Learning Systems

[36] C. Wang, Z. Wang, L. Ma, H. Dong, and W. Sheng, A novel contrastive adversarial network for minor-class data augmentation: applications to pipeline fault diagnosis, *Knowledge-Based Systems*, vol. 271, art. no. 110516, 2023.
[37] C. Wang, Z. Wang, L. Ma, H. Dong, and W. Sheng, Subdomain-alignment data augmentation for pipeline fault diagnosis: an adversarial self-attention network, *IEEE Transactions on Industrial Informatics*, vol. 20, no. 2, pp. 1374–1384, 2024.
[38] K. Wang, K. Li, L. Zhou, Y. Hu, Z. Cheng, J. Liu, and C. Chen, Multiple convolutional neural networks for multivariate time series prediction, *Neurocomputing*, vol. 360, pp. 107–119, Sep. 2019.
[39] W. Wang, L. Ma, Q. Rui and C. Gao, A survey on privacy-preserving control and filtering of networked control systems, *International Journal of Systems Science*, vol. 55, no. 11, pp. 2269–2288, 2024.
[40] X. Wang, H. Liu, J. Du, X. Dong, and Z. Yang, A long-term multivariate time series forecasting network combining series decomposition and convolutional neural networks, *Applied Soft Computing*, vol. 139, art. no. 110214, 2019.
[41] Y. Wang, C. Wen and X. Wu, Fault detection and isolation of floating wind turbine pitch system based on Kalman filter and multi-attention 1DCNN, *Systems Science & Control Engineering*, vol. 12, no. 1, art. no. 2362169, 2024.
[42] Y. Wang, M. Wu, X. Li, L. Xie, and Z. Chen, Multivariate time series representation learning via hierarchical correlation pooling boosted graph neural network, *IEEE Transactions on Artificial Intelligence*, vol. 5, no. 1, pp. 321–333, 2024.
[43] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, Timesnet: Temporal 2d-variation modeling for general time series analysis, In: *International Conference on Learning Representations*, 2023.
[44] H. Wu, J. Xu, J. Wang, and M. Long, Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting, In: *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 22419–22430.
[45] S. Wu, X. Xiao, Q. Ding, P. Zhao, Y. Wei, and J. Huang, Adversarial sparse transformer for time series forecasting, In: *Advances in Neural Information Processing Systems (NIPS)*, vol. 33, 2020, pp. 17105–17115.
[46] J. Xu, H. Wu, J. Wang, and M. Long, Anomaly transformer: Time series anomaly detection with association discrepancy, In: *International Conference on Learning Representations*, 2022.
[47] Y. Xue, M. Li, H. Arabnejad, D. Suleimenova, A. Jahani, B. C. Geiger, F. Boesjes, A. Anagnostou, S. J. E. Taylor, X. Liu and D. Groen, Many-objective simulation optimization for camp location problems in humanitarian logistics, *International Journal of Network Dynamics and Intelligence*, vol. 3, no. 3, art. no. 100017, Sept. 2024.
[48] J. Yang, Y. Yin, L. Yang, S. Ma, H. Huang, D. Zhang, F. Wei, and Z. Li, Gtrans: Grouping and fusing transformer layers for neural machine translation, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 1489–1498, 2022.
[49] Y. Yao, M. Yang, J. Wang, and M. Xie, Multivariate time-series prediction in industrial processes via a deep hybrid network under data uncertainty, *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 1977–1987, 2023.
[50] X. Yi and T. Xu, Distributed event-triggered estimation for dynamic average consensus: A perturbation-injected privacy-preservation scheme, *Information Fusion*, vol. 108, art. no. 102396, Aug. 2024.
[51] X. Yi, Z. Wang, S. Liu and Q. Tang, Acceleration model considering multi-stress coupling effect and reliability modeling method based on nonlinear Wiener process, *Quality and Reliability Engineering International*, vol. 40, no. 6, pp. 3055-3078, May. 2024.
[52] X. Yi, H. Yu and T. Xu, Solving multi-objective weapon-target assignment considering reliability by improved MOEA/D-AM2M, *Neurocomputing*, vol. 563, art. no. 126906, Jan. 2024.
[53] A. Zeng, M. Chen, L. Zhang, and Q. Xu, Are transformers effective for time series forecasting? In: *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, no. 9, 2023, pp. 11121–11128.
[54] J. Zhang and X. He, A partial-label U-net learning method for compound-fault diagnosis with fault-sample class imbalance, *IEEE Transactions on Industrial Informatics*, vol. 20, no. 2, pp. 1798–1807, 2024.
[55] X. Zhang, Y. Lei, H. Chen, L. Zhang, and Y. Zhou, Multivariate time-series modeling for forecasting sintering temperature in rotary kilns using DCGNet, *IEEE Transactions on Industrial Informatics*, vol. 17, no. 7, pp. 4635–4645, 2021.
[56] Y. Zhang and J. Yan, Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting, In: *The Eleventh International Conference on Learning Representations (ICLR)*, 2023.
[57] W. Zheng and J. Hu, Multivariate time series prediction based on temporal change information learning method, *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 10, pp. 7034–7048, 2023.
[58] M. Zhong, X. Zhu, T. Xue, and L. Zhang, An overview of recent advances in model-based event-triggered fault detection and estimation, *International Journal of Systems Science*, vol. 54, no. 4, pp. 929–943, 2023.
[59] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, Informer: Beyond efficient Transformer for long sequence time-series forecasting, In: *Proceedings of the AAAI conference on artificial intelligence (AAAI)*, vol. 35, no. 12, 2021, pp. 11106–11115.
[60] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin, FEDformer: Frequency enhanced decomposed Transformer for long-term series forecasting, In: *Proceedings of the 39th International Conference on Machine Learning (PMLR)*, 2022, pp. 27268–27286.

Chuang Wang received the B.Sc. degree in automation and the Ph.D. degree in petroleum and natural gas engineering from the Northeast Petroleum University, Daqing, China, in 2017 and 2023, respectively.

From August 2022 to August 2023, he was a Visiting Ph.D. Student with the Department of Computer Science, Brunel University London, Uxbridge, U.K. He is currently a Associate Professor with the Artificial Intelligence Energy Research Institute, Northeast Petroleum University, Daqing, China. His research interests include deep learning, transfer learning, and intelligent fault diagnosis.

Zidong Wang (Fellow, IEEE) received the B.Sc. degree in mathematics in 1986 from Suzhou University, Suzhou, China, and the M.Sc. degree in applied mathematics in 1990 and the Ph.D. degree in electrical engineering in 1994, both from Nanjing University of Science and Technology, Nanjing, China.

He is currently Professor of Dynamical Systems and Computing in the Department of Computer Science, Brunel University London, U.K. From 1990 to 2002, he held teaching and research appointments in universities in China, Germany and the U.K. Prof. Wang's research interests include dynamical systems, signal processing, bioinformatics, control theory and applications. He has published a number of papers in international journals. He is a holder of the Alexander von Humboldt Research Fellowship of Germany, the JSPS Research Fellowship of Japan, William Mong Visiting Research Fellowship of Hong Kong.

Prof. Wang serves (or has served) as the Editor-in-Chief for *International Journal of Systems Science*, the Editor-in-Chief for *Neurocomputing*, the Editor-in-Chief for *Systems Science & Control Engineering*, and an Associate Editor for 12 international journals including IEEE TRANSACTIONS ON AUTOMATIC CONTROL, IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, IEEE TRANSACTIONS ON NEURAL NETWORKS, IEEE TRANSACTIONS ON SIGNAL PROCESSING, and IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS-PART C. He is a Member of the Academia Europaea, a Member of the European Academy of Sciences and Arts, an Academician of the International Academy for Systems and Cybernetic Sciences, a Fellow of the IEEE, a Fellow of the Royal Statistical Society and a member of program committee for many international conferences.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNNLS.2025.3542719, IEEE Transactions on Neural Networks and Learning Systems

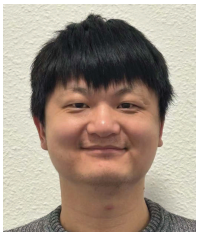
Hongli Dong (Senior Member, IEEE) received the Ph.D. degree in control science and engineering from the Harbin Institute of Technology, Harbin, China, in 2012.

From 2009 to 2010, she was a Research Assistant with the Department of Applied Mathematics, City University of Hong Kong, Hong Kong. From 2010 to 2011, she was a Research Assistant with the Department of Mechanical Engineering, The University of Hong Kong, Hong Kong. From 2011 to 2012, she was a Visiting Scholar with the Department of Information Systems and Computing, Brunel University London, London, U.K. From 2012 to 2014, she was an Alexander von Humboldt Research Fellow with the University of Duisburg-Essen, Duisburg, Germany. She is currently a Professor with the Artificial Intelligence Energy Research Institute, Northeast Petroleum University, Daqing, China. She is also the Director of the Heilongjiang Provincial Key Laboratory of Networking and Intelligent Control, Daqing. Her current research interests include robust control and networked control systems.

Dr. Dong is a very active reviewer for many international journals.

Stanislao Lauria received a Laurea Degree in Physics from the Unversity of Naples Federico II, Italy and the Ph.D. degree in Cybernetics from the University of Reading, U.K. He is a Lecturer at Brunel University London, U.K. He has been research fellow at both the University of Reading and University of Plymouth, U.K. He is a computer scientist with an interest in robotics, HCI, robotics and social media, AI, robotics and education, image analysis, data processing.

Weibo Liu received the B.Eng. degree in electrical engineering from the Department of Electrical Engineering & Electronics, University of Liverpool, Liverpool, U.K, in 2015, and the Ph.D. degree in artificial intelligence in 2020 from the Department of Computer Science, Brunel University London, Uxbridge, U.K. He is currently a Lecturer in the Department of Computer Science, Brunel University London, Uxbridge, U.K. His research interests include intelligent data analysis, evolutionary computation, transfer learning, machine learning and deep learning. He serves as an Associate Editor for the Journal of Ambient Intelligence and Humanized Computing and the Journal of Cognitive Computation. He is a very active reviewer for many international journals and conferences.



Yiming Wang received the Ph.D. degree in the field of computer vision in 2018 from University of Portsmouth, U.K. He is currently a research scientist in the School of Psychology and Neuroscience, University of Glasgow, U.K. His research interests include machine learning, computer vision and human machine interaction.

Futra Fadzil received the Ph.D. in Electrical Engineering and Electronics from Brunel University London, U.K, in 2020. He is currently a research fellow at the College of Engineering, Design and Physical Sciences at Brunel University London, U.K. Over the years, he has gained experience in the power industry and has participated in numerous research projects in the following areas: electrical and instrumentation; operation and maintenance; industrial data acquisition; real-time data analytics; systems engineering; multi-objective optimisation; machine learning; and industrial Internet of Things (IIoT).

Xiaohui Liu received the B.Eng. Degree in Computing from Hohai University, Nanjing, China, in 1982 and the Ph.D. degree in Computer Science from Heriot-Watt University, Edinburgh, U.K, in 1988. He is currently Professor of Computing at Brunel University London where he conducts research in Artificial Intelligence, Data Science and Optimization, with applications in diverse areas including biomedicine and engineering.